Semantics for Interaction Grammars

Guy Perrier, LORIA, Université Nancy2

The formalism of Interaction Grammars (IG) (Perrier, 2002) is devoted to the syntax of natural languages and amalgamates two notions coming from two different traditions in computational linguistics:

- the notion of *polarity* comes from Categorial Grammars (CG) in which grammatical constituents are viewed as consumable resources; expected constituents are marked negatively, available constituents are marked positively and the principle that guides syntactic composition is that positive and negative constituents which are complementary search to cancel each other. IG refine this idea by taking polarities down from the constituents to their morpho-syntactic properties (part of speech, syntactic function, negation ...) in the shape of *polarized features*. Cancellation between opposite features becomes the principle that controls syntactic composition.

- the notion of *tree description* comes from Tree Adjoining Grammars (TAG). The idea is to replace reasoning about complete syntactic trees with reasoning about constraints specifying sets of syntactic trees. Initially, K. Vijay-Shanker used tree description for making the TAG adjoining operation monotone (Vijay-Shankar, 1992), but the interest of the notion goes beyond this application and it is well suited for representing ambiguity and underspecification in an economic way. This gave rise to new formalisms stemming from TAG (Kallmeyer, 1999; Rambow et al., 2001). The basic syntactic objects of IG are also tree descriptions but their specificity is that their nodes are labelled with polarized feature structures.

In natural languages, syntax is a means for accessing semantics and a linguistic formalism cannot deal only with syntax by ignoring semantics. At the same time, the semantic representation is not simply a mirror, a projection or a function of the syntactic representation. It keeps a certain degree of autonomy with respect to the second one; this is fundamental and linguistic formalisms must take this property into account. Unfortunately, in a lot of formalisms, the semantic representation is passively deduced from the syntactic representation. For instance, let us consider CG and TAG which are at the root of IG.

The semantics that is usually associated with CG is Montague's semantics (Carpenter, 1998). Every sentence is interpreted by a formula in higher order logic. This interpretation is a function of the interpretations of the words of the sentence but this function is automatically deduced from the syntactic structure of the sentence. This rigidity is a major default of the formalism: in a certain sense, the whole semantic representation must be already included in the syntactic representation. The latter must be complicated artificially in order to express specific semantic phenomena like quantifier scoping for instance. As a consequence, a sentence gets as many artificial parse trees as interpretations, even if it has a unique actual parse tree.

In the TAG world, we have two ways of representing semantics. We can use Synchronous

TAG (Shieber and Schabes, 1990; Shieber, 1994) or we can see syntactic derivation trees directly as semantic dependency trees (Vijay-Shanker, 1987). Let us consider this last postulate. Even if it is true in a number of cases, its rigidity contradicts reality in several cases. This leads to proposals for modifying the TAG operations so that they be more consistent with semantic interpretation (Rambow et al., 1995).

The purpose of this paper is to add a semantic level to IG in order to get a formalism that captures both syntactic and semantic aspects of natural languages. The fundamental difference with the models of CG and TAG just mentioned is that the semantic level is relatively autonomous with respect to the syntactic level: the mechanism of synchronization between syntax and semantics is as little constraining as possible. Nevertheless, the semantic representation is based on the same notions of polarity and underspecified structure as the syntactic representation, but with a difference: at the syntactic level, we manipulate syntactic trees and syntactic tree descriptions, whereas, at the semantic level, we manipulate semantic directed acyclic graphs (DAG) and semantic DAG descriptions.

1 The level of the polarized syntactic tree descriptions

A syntactic tree description is a set of nodes and parenthood, dominance and precedence relations between these nodes. Nodes represent syntactic constituents and relations express dependencies between these constituents. The morpho-syntactic properties of these constituents are described with feature structures. In this way, a tree description can be viewed as a specification representing a set of syntactic trees and each of these trees can be viewed as a model of the specification.

The originality of IG is to drive syntactic composition by an "electrostatic" mechanism of polarized features. This notion of polarized feature is the continuation of the fundamental idea of CG: capturing the resource sensitivity of natural languages. While a morpho-syntactic feature is usually associated with values in pairs feature-values, a feature in IG is associated with values and polarities in triples feature-polarity-values. Polarity can be -1, 0 or +1, reflecting the fact that a feature is negative, neutral or positive. Positive features represent available resources, negative features expected resources and neutral features linguistic properties that do not behave like consumable resources.

In this way, syntactic composition appears as an "electrostatic" process in which the elementary operation consists in identifying two nodes that bear dual features, that are features with the same name but with opposite polarities. We call this operation *feature cancellation.* By iterating feature cancellations, we specify a syntactic tree description step by step in order to obtain admissible models of it. Such models are syntactic trees that are completely specified with two important properties: neutrality and minimality. They are neutral in the sense that all polarized features have been cancelled and they are minimal in the sense that they contain no additional information with respect to the initial description.

Feature cancellation gives flexibility to the process of syntactic composition by allowing tree superposition, which is important for the construction of lexicalized grammars, but which is forbidden by a lot of formalisms (CG and TAG in particular).

2 The level of the polarized semantic DAG descriptions

A semantic DAG description is a set of nodes and parenthood and dominance relations between these nodes. Nodes represent either predicates or individuals. Parenthood and dominance respectively express predicate-argument and scope relations.

In the same way as at the syntactic level, nodes are labelled with polarized feature structures, which express semantic properties. The same operation of feature cancellation drives semantic composition. By iterating this operation, we can obtain all admissible models of a semantic DAG description. The formalism is flexible enough so that we can encode more or less rich information at the semantic level and we can build a logical representation of an utterance from the DAG expressing its semantic representation.

The semantic representation that we propose here fits in with the new trends in computational semantics, which highlight underspecification (Bos, 1995; Egg et al., 1998; Copestake et al., 1999). The main originality lies in the mechanism of polarities which is used for controlling the process of reducing underspecification.

3 Synchronization between the syntactic and the semantic levels

Synchronization between the syntactic and the semantic levels is performed by a simple function that maps syntactic nodes to semantic ones. This synchronization function is not total because some syntactic nodes, most grammatical words for example, have no semantic representation. It is not injective because two syntactic nodes can have the same semantic representation and, finally, it is not surjective because some semantic nodes have no syntactic antecedents (the predicates that represent quantification for instance).

4 Parsing with IG

IG are lexicalized. A lexical entry associates a word of the language with a syntactic tree description and a semantic DAG description linked together by a synchronization function. In the syntactic description, a node is privileged as its anchor: it represents the position of the word in the syntactic tree represented by the description.

Parsing a sentence with IG consists first in selecting a lexical entry for each of its words. By juxtaposing all selected syntactic tree descriptions we obtain a unique syntactic tree description D_{syn} expressing a syntactic specification of the sentence. The description D_{syn} must be completed with precedence relations between the anchors, which express word order in the sentence. In the same way, by juxtaposing all selected semantic DAG descriptions we obtain a unique semantic DAG description D_{sem} expressing a semantic specification of the sentence. D_{syn} and D_{sem} are synchronized with the union of the synchronization functions from the different lexical entries.

Then, if we assume that the parsing process is driven by the syntax, by iterating feature cancellation inside D_{synt}, we reduce underspecification step by step until we obtain a syntactic tree which is completely specified and in which all polarized features are neutralized.

In this process, the synchronization function plays two roles:

- it contributes to specify D_{sem}: when two syntactic nodes merge in a feature cancellation operation, their corresponding semantic nodes, if they exist, merge at the same time;
- the same mechanism can entail feedback from the semantic level to the syntactic one: if some constraints in D_{sem} forbid the merging of two semantic nodes which is entailed by the merging of two syntactic nodes, both mergings fail.

At the end of a successful parsing process, the description D_{sem} is generally more specific than initially but it can remain underspecified and we have to continue feature cancellations at the only semantic level in order to obtain all semantic representations of the parsed sentence. We can drop this last phase if we want to keep a factorized semantic representation of the sentence.

5 Comparison with other approaches of the syntax-semantics interface

Our model has some similarities with the model of Synchronous TAG (Shieber and Schabes, 1990; Shieber, 1994). Both synchronize two levels of representation, at which the same type of composition operations is used: feature cancellation for IG and adjunction for TAG. Nevertheless, the two models have deep differences. As in basic TAG, the adjoining operation of Synchronous TAG is less flexible than feature cancellation, which allows tree superposition for instance. Synchronization between syntax and semantics is more constraining in TAG than in IG: every adjunction at the semantic level must be linked with an adjunction at the syntactic level so that the syntactic and the semantic derivation trees are isomorphic (Shieber, 1994). This isomorphism is a limit to the expressiveness of the formalism, even if it can be relaxed a bit (Rambow and Satta, 1996). At the opposite, within IG, feature cancellation can occur at the semantic level without affecting the syntactic level.

(Gardent and Kallmeyer, 2003) propose a new semantic construction method for TAG which is not based on derivation trees but on derived trees. Every derived tree is paired with an underspecified logical formula in the shape of a tree description the nodes of which represent conjunctions of predicates. The syntax-semantics interface uses specific features of syntactic nodes which represent semantic individuals or predicates. The unification of these features in the construction of the derived tree contribute to the construction of the logical formula in parallel. This model is very close to our model in its design of the syntax-semantics interface and in the way of representing and solving underspecification at the semantic level: in an underspecified semantic tree, specific nodes are called holes and each one must be plugged with a unique constant node. This mechanism is very similar to our mechanism for neutralizing polarities.

(Muskens, 2001) proposes syntactic representations in the shape of polarized tree descriptions in which polarities are attached at syntactic nodes and not at features. Every syntactic node is labelled with a semantic λ-term à la Montague and the computation of the semantic representation is a mirror of the syntactic computation as in CG but with the same defaults.

The comparison with the approaches based on dependency and unification grammars require more developments and are left for the future.

References

J Bos. 1995. Predicate logic unplugged. In P. Dekker and M. Stokhof, editors, *10th Amsterdam Colloquium*, pages 133–142.

B. Carpenter. 1998. *Type-logical Semantics*. MIT Press, Cambridge, Massachusetts.

A. Copestake, D. Flickinger, and I.A. Sag. 1999. Minimal Recursion Semantics - an Introduction. Draft.

M. Egg, J. Niehren, P. Ruhrberg, and F. Xu. 1998. Constraints over lambda structures in semantic underspecification. In *COLING/ACL'98, Montreal,Quebec, Canada.*

C. Gardent and L. Kallmeyer. 2003. Semantic construction in FTAG. In *EACL'2003, Budapest, Hungary.*

L. Kallmeyer. 1999. *Tree Description Grammars and Underspecified Representations.* Ph.D. thesis, Universität Tübingen.

R. Muskens. 2001. Talking about trees and truth-conditions. *Journal of Logic, Language, and Information,*, 10(4):417–455.

G. Perrier. 2002. Descriptions d'arbres avec polarités : les grammaires d'interaction. In *TALN'02, Nancy, France.*

O. Rambow and G. Satta. 1996. Synchronous Models of Language. In *ACL'96, Santa Cruz.*

O. Rambow, K. Vijay-Shanker, and D. Weir. 1995. D-tree grammars. In *ACL'95, Cambridge, USA*, pages 151–158.

O. Rambow, K. Vijay-Shanker, and D. Weir. 2001. D-tree substitution grammars. *Computational Linguistics*, 27(1):87–121.

S. Shieber and Y. Schabes. 1990. Synchronous Tree-Adjoining Grammars. In *CoLing'90, Helsinki*, volume 3, pages 1–6.

S Shieber. 1994. Restricting the Weak-Generative Capacity of Synchronous Tree-Adjoining Grammars. *Computational Intelligence*, 10(4):371–385.

K. Vijay-Shankar. 1992. Using description of trees in a tree adjoining grammar. *Computational Linguistics*, 18(4):481–517.

K. Vijay-Shanker. 1987. *A Study of Tree Adjoining Grammars.* Ph.D. thesis, University of Pennsylvania.